Automated Neural Flight Control Test (ANCT) Software

What it is: The Automated Neural Flight Controller Test (ANCT) tool is a computational tool that was designed to assist engineers in the analysis of complex systems and to assist with the validation and verification task by helping the engineer conduct, manage and analyze the outputs of test experiments in a simulation environment.

Benefits: Verification and validation of model systems over the entire operating envelope and under specified fault conditions is a difficult and challenging task. ANCT provides a unique set of software tools integrated into the MATLAB environment along with a MySQL database to store and manage all the test results and make it easy for an engineer to use.

Features: ANCT was originally designed to help test engineers validate flight controllers in various flight conditions, quantify performance, and determine regions of stability. It has applications other than flight vehicles, including process control, automotive, medical, etc. Although ANCT was inspired by the desire to create new V&V tools for NN-based aircraft controllers, ANCT is equally applicable to perform V&V on all types of Matlab/Simulink based model systems. Safety–critical applications require that the system can cope with unforeseen catastrophic changes or slow degradation over time, and ANCT can be a valuable tool for assisting engineers in addressing these complex issues.

ANCT was developed in the MATLAB environment and by taking advantage of MATLAB toolboxes, ANCT expands MATLAB to analyze controller models with an assortment of new analysis tools. ANCT can be used to verify and validate Matlab/Simulink models of complex engineering systems. ANCT is designed to analyze a Simulink model of a system by simulating the model using all possible combinations of the model inputs and parameters as determined by the user. By introducing random numbers into the test inputs/parameters, the user can perform Monte Carlo simulation to determine the statistical properties of important output variables as well as the sets of model parameters and inputs that correspond to the system responses that are of interest. In each test case, ANCT simulates the model by using the inputs and parameters from a Test Case Matrix, and evaluates the time-series outputs during a specified time or condition window by using the output evaluation functions. This process yields the Performance score that represents the degree to which an output violates user-defined criteria or failure criteria. ANCT introduces the single measurement score, called the Test score, which is a function of the Performance scores of the various system outputs. The Test score provides a single measurement of the system response in each test case. ANCT uses both a MATLAB MAT-file and a MySQL database to store and manage the test setup information and the test results data. ANCT has three main groups of graphical user interfaces: 1) Test Generation Interface, 2) Simulation Parameters Setup User Interface, and 3) Test Result Manager Interface. The ANCT Genetic Algorithm Analysis tool provides a new approach to the computational analysis of controller models which is faster and more efficient than conventional stochastic simulation methods such as Monte Carlo Simulation.

The purpose of ANCT is not for developing the Matlab/Simulink model, but rather a tool to help test engineers explore the dynamic properties of the models using simulation experiments. The important assumption is that the model itself can be simulated without using ANCT. In order to

simulate a Simulink model, Matlab variables used in the model must be defined and loaded into the Matlab workspace before the execution of the model simulation. The model variables typically are defined using a MAT-file or a M-file. The link to this data is then saved into the model file by using the Preload Function callback. These routines are the standard procedure for developing a Simulink model and each model to be used by a test engineer should already have parameters that have been defined by the model developer. ANCT is not designed to check the compatibility between the model and the model data type. The model developer is responsible for this. ANCT modifies the model parameters between simulation runs and not during a simulation run.

ANCT uses well-defined *inport* and *outport* structures to interface a Simulink model with the ANCT user interface. The advantage of this approach is that ANCT can increase simulation speed by setting all the block parameter to be inline parameters and run the simulation in the *accelerator* mode. This approach also minimizes the number of Matlab commands that are executed during each iteration of a test case. By default, the user can modify many block parameters during a given simulation run. Selecting the inline parameter option makes all parameters non-tunable by default. Making parameters non-tunable allows Simulink to move blocks whose outputs depend only on block parameter values outside the simulation loop. After selecting the inline parameter option, Simulink can speed up the simulation and the execution of code generated from the model. In the *accelerator* simulation mode, ANCT can also change the model inputs and the block parameters between iterations without recompiling the model.

The required Matlab workspace variables must be present in the workspace before the simulation begins. A simple and common way to assign Matlab variables to the workspace before the beginning of simulation is to use the Model pre-load function callbacks in order to extract the necessary information for the block variables, when Simulink is opened.

Consequently, the variables required by the model will be stored inside the function workspace and *whos* command can be used to extract the information about these variables. The most important information about the variable is the variable class. The class of the data determines the range and precision of the stored data. ANCT will allow the user to modify only certain numerical classes of data.
The classes of data that can be modified within ANCT are given in Table 1:

Table 1: Numerical Classes of data that work with ANCT block parameter modification

Name	Description
double	Double-precision floating point
single	Single-precision floating point
int8	Signed 8-bit integer
uint8	Unsigned 8-bit integer
int16	Signed 16-bit integer
uint16	Unsigned 16-bit integer
int32	Signed 32-bit integer
uint32	Unsigned 32-bit integer
logical	Logical variable (0,1)
Simulink.Parameter	Simulink.Parameter Data Object Class

In the Monte Carlo simulation mode of ANCT or in the all combination test input mode, the user can generate a test case matrix that consists of the inputs given to the model through *inports* for all the test cases. For the iterations associated with a given test, ANCT uses the Matlab command, *sim*, to run the model simulation. The test case inputs are assigned to the *sim* command by using an external input argument.

For information, contact:

Fola Soares (Primary Point of Contact)
Contek Research, Inc.
Tel: 310-414-6720, e-mail: fola@contekresearch.com
At NASA DFRC: tel: 661-276-5536, email: fola.soares@dfrc.nasa.gov